Towards Next-Generation Association Studies:

Complex (pan)genotype-phenotype mapping with deep neural networks

Divyae Kishore Prasad

5th March 2020

Abstract

Genome-wide association studies (GWAS) are the gold standard for genotype to phenotype mapping effort. They have aided in the understanding of variations in microbial genomes, such as those associated with antimicrobial resistance. However, current microbial GWAS struggle to achieve both statistical power needed for confident association calling, as well as the precision required to reject spurious findings (false-positives) (Earle et al., 2016). To this end, we employ Jaillard et al. (2018)'s tool `DBGWAS` for reference-agnostic variant-calling, and use the consequently obtained pangenome graph as inputs to deep neural networks (DNNs). The pangenome graph, along with the antimicrobial response phenotype, is employed for building genotype-phenotype mappings via DNNs. We hypothesize that the DNNs could learn the interactions between polymorphisms- a subset of those belonging to true epistasis in microbial genomes. We introduce `PANet` (PAngenome Network, **Fig. 7**) for classifying antimicrobial response on a panel of 280 *Pseudomonas aeruginosa* genomes (**Fig. 6**) that are relevant for cystic fibrosis. While the underlying decision-making mechanism of DNNs is not completely understood, a comparison of `PANet` performance to elastic net linear regression reveals that our DNNs can perform as well as linear models (for the given panel). The receiver operating characteristic (ROC) curves of both `PANet` (**Fig. 12**) and elastic net linear regression (**Fig. 13**) are highly similar, suggesting that `PANet` could capture minimal epistasis. The area under these ROC curves is also the same. When taken together, these unanticipated results could imply that antimicrobial resistance genotype-phenotype mapping for the given panel might indeed have a linear relationship. Previously, genotype-phenotype mapping literature such as Bellot et al. (2018) and Romagnoni et al. (2019) have also reported that their DNNs were as competitive as linear models. Nonetheless, DNNs were able to identify more variants with small effect-sizes as compared to linear models. A future experiment could entail the discovery of salient features important for classification. And therefore, assist in establishing that DNNs could indeed help uncover more polymorphisms (particularly those with small effect-sizes) associated with complex phenotypes (**Sec. 4.2**).

Keywords – genotype-phenotype mapping, DNNs, machine learning, antimicrobial resistance (AMR), epistasis, genome-wide association studies (GWAS)

EXAMINER: **Dr. Aldert Zomer** (A.L.Zomer@uu.nl)

SUPERVISORS: **Dr. Marleen Balvert** (M.Balvert@tilburguniversity.edu)

Prof. Dr. Alexander Schönhuth (A.Schoenhuth@cwi.nl)

Towards Next-Generation Association Studies:

Complex (pan)genotype-phenotype mapping with deep neural networks

Minor internship performed at:

Centrum Wiskunde & Informatica, Amsterdam 1098 XG, The Netherlands *&*

Theoretical Biology & Bioinformatics, Utrecht University, Utrecht 3584 CH, The Netherlands

[‡]Correspondence: div.prasad03@gmail.com

[†]*Submitted to Universiteit Utrecht for partial fulfilment of M.Sc*

Table of contents

List of figures

List of tables

1 Introduction

Thanks to advances in Next-Generation Sequencing technologies, we are presented with vast quantities of genomic data. While we are beginning to unravel the secrets to evolutionary forces hidden 'encrypted' within the genetic code of life, subsequent advances in computational approaches are needed to stay abreast with the so-called tsunami of biological big-data. Genome Wide Association Studies (GWAS) are the current gold standard for gaining insights into the genetic factors underlying the phenotypic properties of organisms. Originally developed for human genomics for exploring polymorphisms in coding or non-coding (such as regulatory elements) that are associated with risk for certain diseases, GWAS in recent years has been suitably adapted for microbial genomics. Gaining a fundamental understanding of forces in evolution, e.g., under strong evolutionary selection forces such as drug-resistance, virulence, and pathogenicity, is probably one of the most studied aspects in microbiology and microbial evolution. Steps are needed for successfully bridging knowledge gaps in the interpretation of sequencing information. Gaining biological insights will aid in translating knowledge into scientific and clinical practice. Discovery of new genomic or molecular markers will not only be of great epidemiological relevance (e.g., improving clinical practice guidelines), but such methods can also be applied to investigate eco-evolutionary aspects of microorganisms.

Although GWAS has helped identify genomic variations (polymorphisms in genes, transcription factor binding sites, or non-coding regions, etc.), in the discovery of alleles associated with the phenotype of interest, GWAS still has some limitations when applied to microbial genomes (Collins and Didelot, 2018). In recent years appropriate measures to overcome confounding factors such as strong population structure (Earle et al., 2016) have made improvements in microbial GWAS and thus successfully curtail the rate of false positives. Apart from striking differences in genome complexity between microbes and higher eukaryotes, phenotypic and corresponding genotypic evolution is driven primarily by the high rate of horizontal gene transfer (HGT) (Sakoparnig et al., 2019; Lin and Kussell, 2019; Dixit et al., 2017), and large effective population size. Unsurprisingly, strains within a microbial species can often be highly divergent. Some of these challenges may be overcome by using reference-agnostic algorithms (Jaillard et al., 2018) for variant-calling, such as pangenome graphs (Marschall et al., 2016). Nonetheless, while the cost and effort for sequencing and obtaining the genetic blueprint of microbes has dropped manifold, our computational approaches are yet to keep pace with the advancements. Analysis of genomic variability often lacks statistical power, as genetic differences between strains often contribute substantial phenotypic variability (Earle et al., 2016; Cordero and Polz, 2014). To summarise, the genotype-phenotype mapping problem is severely statistically underpowered, while at the same time is confronted by the 'curse of high dimensionality.' In **Sec. 3.1** we describe approaches undertaken to overcome them.

In this thesis:

We discuss the biology of antimicrobial drugs, their targets (**Sec. 1.1**), and corresponding computational challenges pertaining to biomarker discovery in microbial GWAS (**Sec. 1.2**). Next, we describe the computational pangenomics (Marschall et al., 2016) approach for obtaining reference-agnostic variant-calling, using the recently published method (Jaillard et al., 2018). Using Jaillard et al. (2018)'s tool `DBGWAS` (de Bruijn Graph GWAS), we obtain a description of the entire pangenome in a single graph data structure as a compacted de Bruijn Graph (cDBG), wherein unitigs (or unique extended kmers as variants) are the nodes of the cDBG, and the paths along different nodes (that describe the sequences of different strains in the pangenome) are represented as edges.

Now, using the nodes of the cDBG as input to neural networks and phenotypes as output, we develop a deep neural network (DNN) architecture for genotype-phenotype mapping. We highlight the major advantages of using DNNs for genotype-phenotype mapping, as compared to traditional linear methods. Additionally, we describe the rationales employed in developing deep convolutional neural networks. Thereafter, the cross-validation employed for hyper-parameter tuning of the DNNs is illustrated (**Fig. 10**), which ultimately lead to the final architecture of `PANet` (**Fig. 7**). Next, the final model training on the entire dataset (**Fig. 11**) is shown to ensure overfitting. In **Sec. 3.6**, the `PANet` model we report unbiased performance estimates of both balanced and imbalanced testing.

Additionally, we compare the `PANet` model performance to a logistic regression with elastic net penalty (weighted combination of L1 and L2 regularization, in **Sec. 3.7**). In the discussion (**Sec. 4**), we discuss the results and present outlook for future work on genotype-phenotype mapping, before concluding the key results of this project.

1.1 Antimicrobials & their molecular targets

Antimicrobial drugs in the previous century have been one of the contributory factors in improving life expectancy. However, in 2014, the World Health Organization recognized the rise of Antimicrobial Resistance (AMR) amongst pathogens, 'a serious threat to global health and food security' (WHO, 2014). Hence development methods to discover molecular targets underlying molecular evolution in the context of AMR are of paramount importance.

In this thesis, we focus on investigating resistance and sensitivity of *Pseudomonas aeruginosa*, an multi-drug resistant (MDR) pathogen (van Belkum et al., 2015; Dunne et al., 2017), including the antimicrobial Amikacin (which belongs to the aminoglycoside class). Amikacin is delivered via injections, either into muscles or veins, and it works by irreversibly binding to 16s rRNA and 30s subunit of prokaryotic ribosome (Doi and Arakawa, 2007). After that, it changes the ribosome's shape such that it cannot correctly read mRNA codons, hence hindering new protein synthesis (Jaillard et al., 2017).

P. aeruginosa has remarkable genotypic plasticity, and mapping of its intrinsically advanced antimicrobial and multi-drug resistance mechanisms is poorly understood. *P. aeruginosa* has a highly plastic pangenome, and it can gain and lose a large repertoire of dispensable genes to suit its environment. For microbes, a large gene repertoire has been postulated to aid an organism's ability to adapt to its changing niche robustly. Such reasoning has been hypothesized to be the driving factor behind the pathogenesis of refractory microbes (van Belkum et al., 2015) such as *P. aeruginosa*. The notorious gram-negative bacterium *P. aeruginosa* has versatile metabolism (Jaillard et al., 2017; Jansen et al., 2016; van Belkum et al., 2015), that enables it to thrive in diverse natural ecologies and nosocomial (in hospital) environments. While *P. aeruginosa* infections in healthy people do not pose serious problems, pathogenesis in (Immuno)compromised patients manifests as opportunistic (Stover et al., 2000), wherein the infections co-occur in existing illnesses, most notably in cystic fibrosis, HIV or cancer patients undergoing chemotherapy (Wagner and Iglewski, 2008). In particular, the respiratory tract is most frequently infected. It often presents as acute ventilator-associated pneumonia or chronic destructive lung diseases in patients with cystic fibrosis in intensive care units of hospitals (Pier, 2005). Within such immunocompromised patients of cystic fibrosis, the resilience of *P. aeruginosa* to existing therapeutic interventions, including antimicrobials of last resort, makes *P. aeruginosa* infamous for aggravating clinical complications.

1.2 Challenges in microbial GWAS

For adequately adapting microbial GWAS methods, two key challenges need to be addressed. Firstly, staggering rates of HGT in ecological niches contribute to rapid rates of microbial evolution. Fast evolution may manifest as lack of sequence similarity either through homology (shared common ancestry) or homoplasy (convergent sequence evolution) in the microbes being investigated, and thus statistically underpowering GWAS. Secondly, differences amongst strains account for a large proportion of genotypic and hence phenotypic variability (Earle et al., 2016; Cordero and Polz, 2014). When the above two aspects are taken together, the true causal variants may be confounded by strong population structure, and strong linkage disequilibrium (LD). Statistically, resolving these challenges is not trivial (Earle et al., 2016). Association studies in *P. aeruginosa* are challenging in microbial panels for several reasons:

1. high genome plasticity from HGT resulting in a long-tailed distribution of rare genetic variants (Sakoparnig et al., 2019; Dixit et al., 2017)
2. strong population structure in geographically distinct lineages
3. strong LD across regions within genes or across the length of the genome
4. lack of sequence similarity, possibly indicating that many genetic variations could bring about phenotypic states

Linear mixed models in microbial GWAS

In a seminal paper, (Earle et al., 2016) showed that using linear mixed models (LMM) that correct for population structure improves statistical power in GWAS. Regression on a phenotype arises from both the fixed (β) and random effect components (α and ε). In LMM (Hoffman, 2013), the first term is the regression on the unitig locus and models the fixed effect contribution (β) of a specific presence-absence set of polymorphism ($X_i^\top$) towards the phenotype(Y_i). The design matrix (W) models the covariance (can be corrected for population structure by removing the first two principal components), and hence takes into account the relatedness between the strains. The random-effects comprise of both the random effect of the background locus (α) as well as the noise (ε) that is intrinsic to the phenotype of each strain (not directly explained by the model). Hence, the phenotype (Y_i) for each of the samples (n) can be regressed with **Eq. 1**.

$$Y_i = X_i^\top \beta + W_i^\top \alpha + \varepsilon_i, \quad i = 1, \ldots, n \tag{1}$$

The entire regression model for all the n samples boils down to solving for fixed-effect and random effect regression coefficients, β and α respectively (**Eq. 2**). As the random effect coefficient, α is the random complement to the fixed effect β, and it follows a normal distribution of the variance of the design matrix W and hence is centred at zero (the mean is 0).

$$Y = X\beta + W\alpha + \varepsilon \tag{2}$$

Graph representation of pangenomes

Graph representations of genomes are only recently beginning to be used in bioinformatics. E.g., excellent and novel graph-based tools such as `vgtools` (Garrison et al., 2018), provide a toolbox for pangenome graph construction. Graph-based pangenomes are a first step in improving the biologists can look at genotypic and corresponding evolutionary landscapes. Because traditional variant-calling (identification of variations from genome sequences or other fragments of sequencing data) relies on mapping to a reference, thereby introducing biases towards the choice of reference. While the graph genomes are beginning to gain traction in scientific communities, thanks to efforts from (Marschall et al., 2016), they are not yet a standard practice in computational biology and bioinformatics research.

We employ another excellent method for loss-less pangenome representation, namely the recently published tool DBGWAS (Jaillard et al., 2018). It allows investigation of genomic

variations associated with phenotypes such as antimicrobial response, on a panel of closely related microbes and observed phenotypes. First, DBGWAS constructs a de Bruijn Graph (DBG) of given kmer size, representing different genomic fragments of length 'k' as nodes and adjacent nodes share (k-1) length of nucleotides between them (**Fig. 1a,b**). Next, the DBG is compacted in a loss-less manner (**Fig. 1c**), to form unitigs (analogous to polymorphisms). By extending kmers, until they appropriately represent similarities and differences in shared genomic regions and variable regions, respectively, we obtained unitigs.

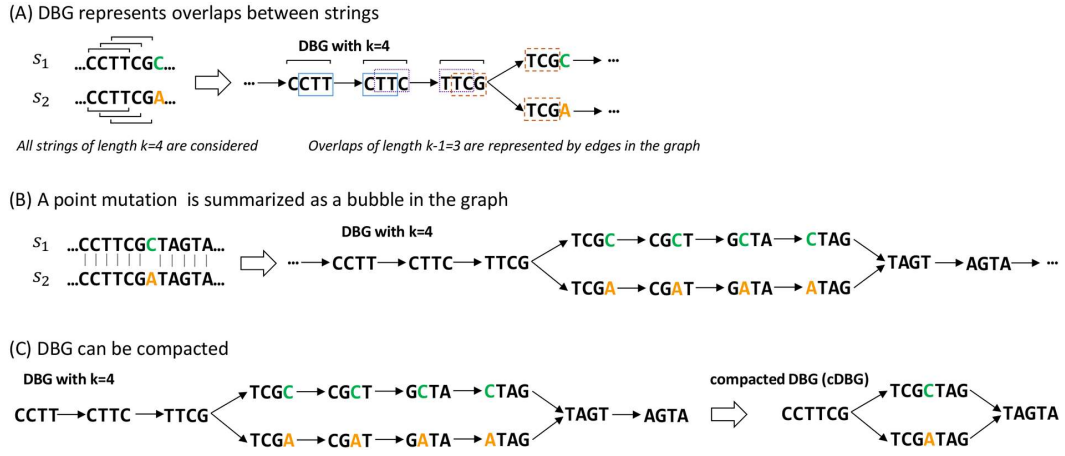


Figure 1: Compacted de Bruijn Graph: The unitigs (variable length kmers) are the nodes of cDBG. Unitigs collapse redundant (polymorphisms in perfect LD) genomic fragments into single nodes (compaction process). They preserve local polymorphisms within a unitigs, under the assumption that each unitig can undergo HGT as one unit. The presence or absence of unique unitig pattern is given as the input to deep neural networks, e.g. PANet (**Fig. 7**). Figure taken from Jaillard et al. (2018).

An important hyperparameter choice that governs the structure of the pangenome graph is the kmer length that initializes the DBG construction and determines the subsequent distribution of unitig lengths. Towards smaller values of k, the graph contains some unitigs that repeat at different regions in the genome, resulting in a graph containing many cyclic components. Cyclic components are undesirable in such a graph-based pangenome, as it leads to loss of specificity for a repeating unitig. Upon increasing the kmer length, the graph becomes increasingly disentangled. As a trade-off between non-specific unitigs (hence variants) and risk of unitigs being coalesced together, Jaillard et al. (2018) show an appropriate kmer choice (**Fig. 2**).

The Receiver operating curve or ROC curve illustrates the diagnostic capability of binary classifiers. As the discriminative threshold is varied, the true-positive rate (TPR) and false-positive rate (FPR) is obtained at the different threshold settings. This estimation quantifies the rewards against the risk; TPR is the sensitivity or probability of detection, while FPR is the probability of false alarm (1-specificity), respectively. The line $y = x$ signifies a random guess, with 0.5 as AUROC for a random classifier (no pattern recognition within a balanced training-data), and AUROC of 1 as a perfect binary predictor. The sample space of AUROC between 0.5 and 1 encompasses all real-world classifiers, with exponentially increasing performance. I.e. it is much easier to increase AUROC from 0.6 to 0.65 than to have an increment from 0.9 to 0.95.

Note that in the ROC curve **Fig. 2**, Jaillard et al. (2018) are asking a different question from

what we investigate. Jaillard et al. (2018) were interested in quantifying correct classifications 'significantly' associated variations, with the variants labelled as true if they are described in published literature. As we ask how well can we predict the two phenotypes apart, our results (**Fig. 12**) are not directly comparable with theirs.

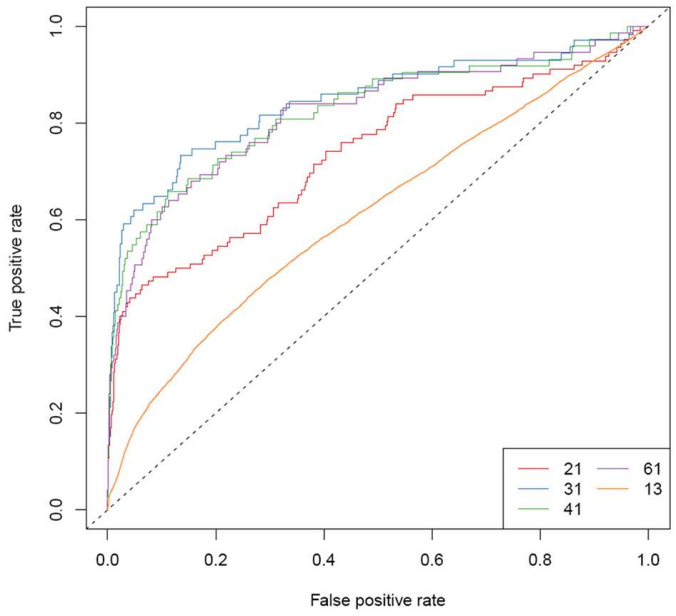


Figure 2: Choice of initial word lengths k: Here, Jaillard et al. (2018) perform statistical association testing to assess if the LMM (bugwas implementation) can detect true variants- true variants are those in the reference databases. They show that the area under the ROC curve (AUROC) is maximum for $k = 41$, and we also use the same. Figure taken from Jaillard et al. (2018).

Jaillard et al. (2018) performed association testing using the presence-absence pattern of unitigs, with linear mixed models that correct for population structure. In this work, we employ the same presence-absence unitig matrix as input to neural networks. Currently, we ignore the covariance that describes the inter-dependencies of strains across all the unitigs and describe the proposed follow up experiments in the Outlook (**SubSec.4.2**).

1.3 Artificial neural networks: going deep

In machine learning, algorithms are given training examples that are described by potential features (such as presence and absence of unitig - genotype) and with known properties (such as antimicrobial response - phenotypes). With sufficiently rich and diverse training examples, algorithms can generalize the relationship between the input features and output phenotypic properties.

Within machine learning, artificial neural networks comprise of neurons as computational units (**Fig. 3**). While neural networks have long been recognized as universal approximators of arbitrary functions (Hornik, 1991; Leshno et al., 1993), it is only in the past decade that we have witnessed a revolution, wherein algorithms can interpret and perform tasks that were

once considered impossible like self-driving vehicles, including the development of the underlying mathematical theory (Lu et al., 2017; Hanin and Sellke, 2017). Few primary reasons for this revolution are substantial gain in computational power from massive parallelization on graphical processing units (GPUs), but also the development of so-called appropriate network architectures. E.g., DNNs - a variation of neural network architecture comprising many hidden layers, can perform equivalently to shallow networks but with exponentially lower model complexity (fewer number of parameters that need to be estimated during training) (Simonyan and Zisserman, 2014). Deep learning has thus far provided great performance improvement in image and speech recognition, natural language processing, and more recently is in computational biology and bioinformatics.

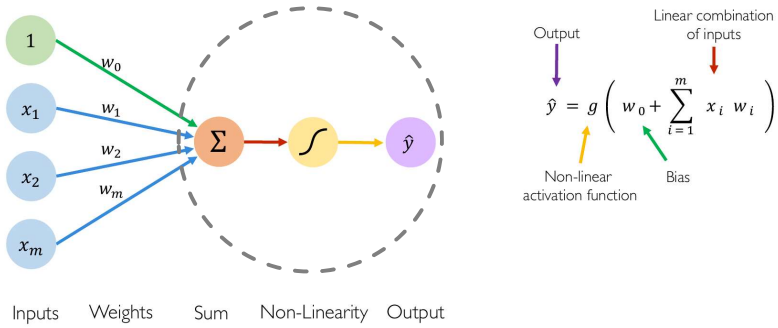


Figure 3: Neuron: a functional unit of neural network A single computational unit within a neural network is a neuron- it combines the inputs through linear transformation (via weights) and offsets it with a bias term. This linear combination of all the inputs is transformed with a suitable non-linear activation function, thus successfully mapping (or transforming) the m-dimensional inputs space onto a n-dimensional output space. For a single neuron, the output dimension $n = 1$. Figure is adapted from (Amini, 2020).

Briefly, (deep) neural networks take raw and minimally processed features as input and transform the raw features into an increasingly abstract feature representation as we proceed along the hidden layers of the network. By combining the outputs from preceding layers followed by a non-linear activation function such as rectified linear unit (ReLU), it can encapsulate highly complex functions(Zhang et al., 2019; Pérez-Enciso and Zingaretti, 2019). Because of their inherent representational richness, DNNs can capture dependencies in the input unitigs and their interaction effects, especially if they are non-linear. Neural networks, if sufficiently deep, can universally approximate functions of an arbitrarily large dimensional space, and are an excellent choice for handling 'the curse of dimensionality'. As the pangenome graph of *P. aeruginosa* which we describe in **Sec. 1.2** and **Fig. 4a** has more than a million input nodes, DNNs are well-suited for genotype-phenotype mapping. Currently, a predominant type of neural networks is deep convolutional neural networks. Here, convolution filters are used for perfectly capturing associations between the input variables and are alternatively used for reducing the number of parameters that need to be estimated. We employ DNNs for genotype-phenotype mapping as they have the potential to outperform existing linear methods.

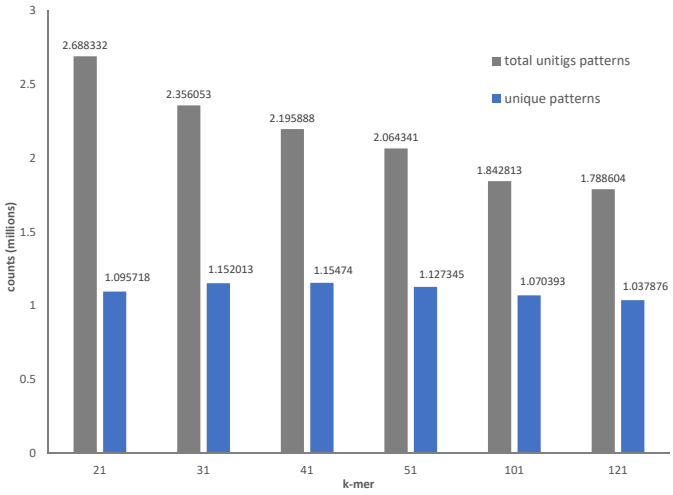


Figure 4: Variation in cDBG with kmer length For each kmer length, total unitig patterns and unique patterns were obtained. While the unitigs become longer, the number of 'unique' patterns of unitigs (relevant as the input neurons of DNNs) does not change in count, while the total number of unitig patterns decrease slightly in count. Interestingly, 41 is the kmer size for which most unique patterns are present, and was also used by Jaillard et al. (2018). Unitig distribution obtained by varying kmer length during graph construction via DBGWAS.

In particular, the effect of genetic interactions, although which is known to be pervasive in biological systems (Pedruzzi et al., 2018; Mackay, 2014), the effects of potential epistasis in genotype-phenotype mapping remain vastly understudied. We hypothesize that deep neural networks can pick up additive and often non-linear interactions between genetic polymorphisms associated with a phenotype. Consequently, DNNs could outperform linear models that seek to understand complex genetics underlying phenotypes. Currently, studies involving DNNs in microbial genomics are limited and have only addressed typical classification problems like binning of metagenomic reads into operational taxonomic units (OTUs) (Fiannaca et al., 2018; Wassan et al., 2019). We shed light on the debated applications of deep neural networks black-box to 'understand' genomic data in computational biology.

Contribution

Traditional approaches in GWAS have focused on the enrichment of single polymorphic regions, that differ in statistical abundance between cases (resistance phenotype) against controls (sensitive phenotype). While there are no reasons to believe that enrichment of multiple genes and their interactions could not bring about phenotypes, most of previous GWAS literature has not looked at the effect of interactions, and hence ignoring the role of epistasis in genotype-phenotype mapping. Epistasis is formally defined as the interaction between individual polymorphisms, and broadly comprises of additive, dominant, and non-additive (and non-linear) effects of interaction (Mackay, 2014). Working under a simplified assumption that DNNs can capture genetic interaction patterns within the genotypic landscape (as described by, e.g. pangenome graph) underlying a phenotype, we hypothesize that they can outperform linear models, as existing linear methods in GWAS cannot capture complex interactions. If this indeed is the case, performance gains must be observed when DNNs are used for genotype-phenotype mapping. We test the validity of this hypothesis in the project.

2 Methods

Our approach has a two-fold novelty, namely the advantages of reference-agnostic graph pangenome, and leveraging universal approximation power of DNNs. To the best of our knowledge, this project is the first to employ deep learning in microbial GWAS. We hope that methodological breakthroughs on similar lines would bridge knowledge gaps about advantages, practicality, and feasibility of DNNs in (microbial) genotype-phenotype mapping.

Panel of *P. aeruginosa*

A panel of 280 *P. aeruginosa* was obtained from Jaillard et al. (2018), wherein 47 strains were resistant to Amikacin. On this entire set of 280 strains, a cDBG was constructed, using the DBGWAS tool, with kmer length of 41 as a parameter. The graphical methodological summary illustrates the key steps of the project:

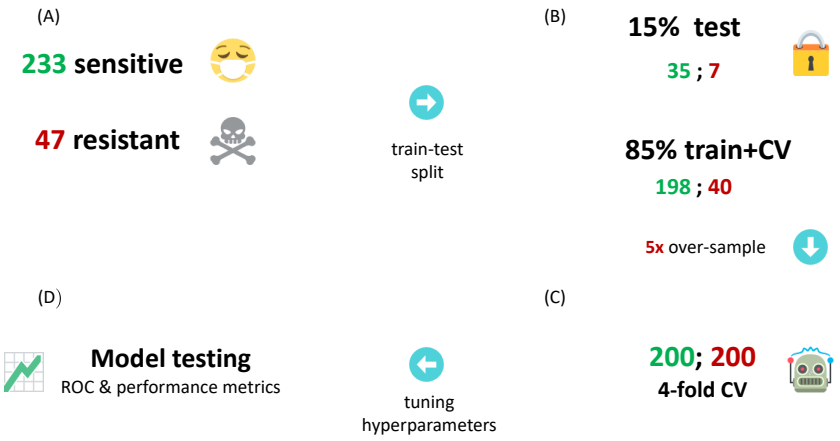


Figure 5: Graphical summary of methods From the 47 resistant and 233 sensitive genomes in the panel (A), 85% is used as the training-data, and the remaining 15% is locked away as the test dataset during hyperparameter optimization (C). 188 (out of 233) resistant, and 188 (4-fold oversampling from 47) sensitive are sampled to obtain a balanced class (B). We design the architecture of the DNN and tune its hyper-parameters using a 4-fold cross-validation. A 4-fold cross-validation comprises of 4 rounds, wherein for each round training is performed on 3 parts of training-data and validated on the remaining part. DNN architectures were assessed by monitoring the prediction accuracy on the left out validation set. The model testing (D) is only performed twice in the thesis: once after fixing the architecture and testing PANet (**Fig. 12**) and the other for testing the (hyperparameter tuned) elastic net linear regression (**Fig. 13**).

2.1 Pangenome analysis

The pangenome of the panel of *P. aeruginosa* was analysed using `Roary` (Page et al., 2015). Such pangenome analyses can provide an overview of the core, and accessory genes of the panel, and at the same time provides a coarse-grained approximation (gene-level) of the relatedness of microbes in the panel. The gene prediction algorithm via `Roary` looks for open reading frames, as well as considers ab-initio predictions. Once gene sequences are obtained, gene clustering is performed with 95% sequence similarity over 95% length of sequence coverage as the cutoff for defining gene clusters.

Variation graph (vg) experiments

After cDBG construction, we extensively experimented with the ordering of unitigs as it could be important, e.g. (speculatively), enabling the biologically meaningful convolution operations. Algorithmic implementations such as in variation graph `vg` (Garrison et al., 2018) try generating approximate (or greedy) solutions in polynomial time. Despite our efforts, experimenting in `vg` for graph sorting as a directed acyclic graph (DAG)) did not obtain successful results. To the best of our knowledge, no algorithm or method exists that constrains spatial structure (formally known as partial ordering in graph theory) in polynomial time.

2.2 Building neural networks: going deep

The genomic dataset is reasonably large, memory-wise. So much so that the more interpretable machine learning methods such as Random Forest could not handle memory constraints or other bottlenecks presented by the pangenome graph. Attempts to build these machine learning models in the programming language R crashed within a few minutes of computation. To the best of our knowledge, DNNs have not been applied for mapping the microbial phenotypes from purely genomic information.

Inspirations for network architectures were drawn from the early 'very deep networks' from 2014-15, such as VGG-16 and VGG-19 (Simonyan and Zisserman, 2014). After that, more sophisticated networks have also been introduced, such as ResNet (He et al., 2016), U-Net (Ronneberger et al., 2015), and Inception (Szegedy et al., 2015). Additionally, we did not run into the issue of vanishing gradient (He et al., 2016; Pan et al., 2016), which wherein, e.g. ResNet or U-Net also incorporate residual-blocks to improve training. Currently, it is unknown if similar networks with residual-blocks. Convolutional network architecture choices were experimented with by varying receptive field length (size of convolutional filter), stride length (jumps performed along the hidden layer neurons while performing convolutions), the number of convolutional blocks, dropout rate, and the number of fully connected neurons. We start with a simple adoption of VGG-16 network and build models. Heuristic driven hyperparameters tuning and making appropriate changes in architecture)in deep learning involves trial and error process, and may usually be done manually.

For learning the parameters (convolutional kernels, weights and biases of FC layers; e.g. **Tab.2**) in DNN models via back-propagation optimization, binary cross-entropy (**Eq. 3**) was employed as the loss function. The cross-entropy loss quantifies the total divergence (deviation) of the binary predicted indicator of classification (q, and $1-q$; correct classification is 1, incorrect is 0) from the probability predicted by the neural network (p, and $1-p$; probabilities are continuous real values between 0 and 1) respectively (**Eq. 3**).

$$loss = -(q \cdot \log(p) + (1-q) \cdot \log(1-p)) \tag{3}$$

Further, the DNNs can be trained by stochastic computation of gradients on minibatches (with respect to loss function) and subsequent loss optimization. Several adaptive solvers available in the Keras implementation of TensorFlow were experimented with, such as Adamax, Adam, Nadam, AdaDelta and as well as the stochastic gradient descent (SGD). Notably, the standard stochastic gradient descent SDG did not train at all. Although the adaptive solvers should be

similar in theory, this was not the case for our experiments. For all the cross-validation runs, the Adamax optimization algorithm had the best performance.

With the presence-absence descriptions pattern of unitigs in the cDBG and the antimicrobial (sensitive versus resistance) response, we experiment building DNN architectures that can approximate the genotype-phenotype mapping across all strains in the pangenome. As a starting point, we took inspiration from VGG-16 and VGG-19 (Simonyan and Zisserman, 2014) network architectures. These 'very deep convolutional networks' were the state of the art that won first and second places in localization and classification respectively in the 2014 ImageNet Large Scale Visual Recognition Challenge (ILSVRC), one of the most prestigious computer vision competitions. Simonyan and Zisserman (2014) demonstrated that substantially increase the depth of convolutional network architectures can tremendously increase classification performance. In their VGG networks, a dimensionality reduction is typically performed to obtain representational feature maps, before introducing computationally expensive fully connected operation towards the end of neural networks. For dimensionality reduction and finding non-linear interactions, convolutional filters were used to transform (convolution function with ReLU activation) the input pattern of unitigs onto increasingly abstract feature maps. Along the depth of DNN, layers of the feature maps capture dependencies between the preceding layers. Note that convolutional filters are often used in signal processing and can perfectly capture correlations between dependencies.

2.3 GPU computations

Performing computations on Graphical Processing Units (GPUs) allowed us to exploit t massively parallelization of SDG optimization for DNN parameter training. Model training on GPUs, courtesy the High Performance Computing cluster at UMC Utrecht, led to almost 8-fold improvement in neural network computations when compared on CPUs.

All of the model training, and cross-validation was performed on the Tesla P100 16GB GPU running NVIDIA-SMI driver v390.30 and CUDA v9.1.85 on Linux x86 64-bit systems via CentOS 7. Python v3.6.8, and TensorFlow along with the following major computational dependencies were employed for computational experiments: tensorflow-gpu v1.12.0, keras v2.2.4, numpy v1.15.4, matplotlib v3.0.2, cudatoolkit v9.0.0, cuDNN v7.3.1, cuda v9.0.0, scikit-learn v0.20.2.

To aid reproducibility, we provide the entire kernel environment as '.txt' file in addition to an Anaconda '.yaml' file that specifies all dependencies. The entire workflow has been catalogued in a series of Jupyter notebooks, some of which are maintained on GitHub (Prasad, 2019).

Finally, we intend to open source this project with all pertinent codes upon acceptance.

3 Results

In **Sec. 3.1**, the genome plasticity is visualized for the panel of *Pseudomonas aeruginosa*. Next, we show the cross-validation (CV) procedure in **Sec. 3.2**, that allows tuning of the hyper-parameters and selection of the best performing network architecture. We show the network architecture of `PANet` (**Fig. 7**). Thereafter, the final training of the best network architecture model on the entire training-dataset (**Fig. 11**) was performed with the most suitable (in the space that we looked at) hyper-parameters. Finally, we show the testing (**Fig. 12**), the estimates of performance metrics (**Tab. 3**), and the confusion matrix (**Tab. 4**).

With the same training and testing splits, model fitting on elastic net training-dataset was performed, including a search for the choice of hyperparameters for the elastic net. By having two hyperparameters for both the L1 and L2 norm, elastic net embodies best of both the regularisations. L1 norm penalty is known to introduce sparsity and hence performs feature selection. Using the same CV scheme as described (**Fig. 5**), elastic net hyperparameters were suitable tuned. Results of the final testing procedure of elastic net are shown in **Fig. 13**.

Comparison of both the models reveals several exciting aspects of genotype-phenotype mapping for the given problem, as well as the machine learning classification task.

3.1 Pangenome of *Pseudomonas aeruginosa* is highly plastic

Majority of the genes in the *P. aeruginosa* panel are accessory (**Tab. 1**). The microbial species maybe towards the upper-end of genome flexibility, especially when comparing pathogenic microorganisms.

Table 1: Flexibility of the PA pangenome Genomes of different strains can be highly variable, wherein only 15 % core genes are common across all strains. Majority of the gene clusters are accessory, as they are not shared by all strains in the pangenome, hence variant-calling by aligning individual strains to reference genome was not a viable option.

Genes	Abundance	Gene clusters	
		Number	Percentage
Core	(99% <= strains <= 100%)	2729	8.17 %
Soft-core	(95% <= strains < 99%)	2136	6.39 %
Shell	(15% <= strains < 95%)	2072	6.20 %
Cloud	(0% <= strains < 15%)	26483	79.24 %
Total	(0% <= strains <= 100%)	33420	

The genomic flexibility of the *Pseudomonas aeruginosa* panel is visualized in **Fig. 6**. Gene content may be a phylogenetic signal, but even within this smaller presence-absence pattern, one could appreciate a few properties of the binary matrix. For the matrix, presence-absence pattern is symmetric, i.e. by swapping the presence with absence (vice-versa is equivalent), the important signals (for pattern recognition or classification) do not change. Also, the highly abundant or highly rare gene clusters would not be informative. Only some narrow

middle band of gene clusters (where a strong variation can be seen) would be meaningful for distinguishing the sensitive from the resistant phenotypes, if gene clusters would be employed for classification, instead of unitigs. Effectively, even this simpler problem would be greatly statistically underpowered.

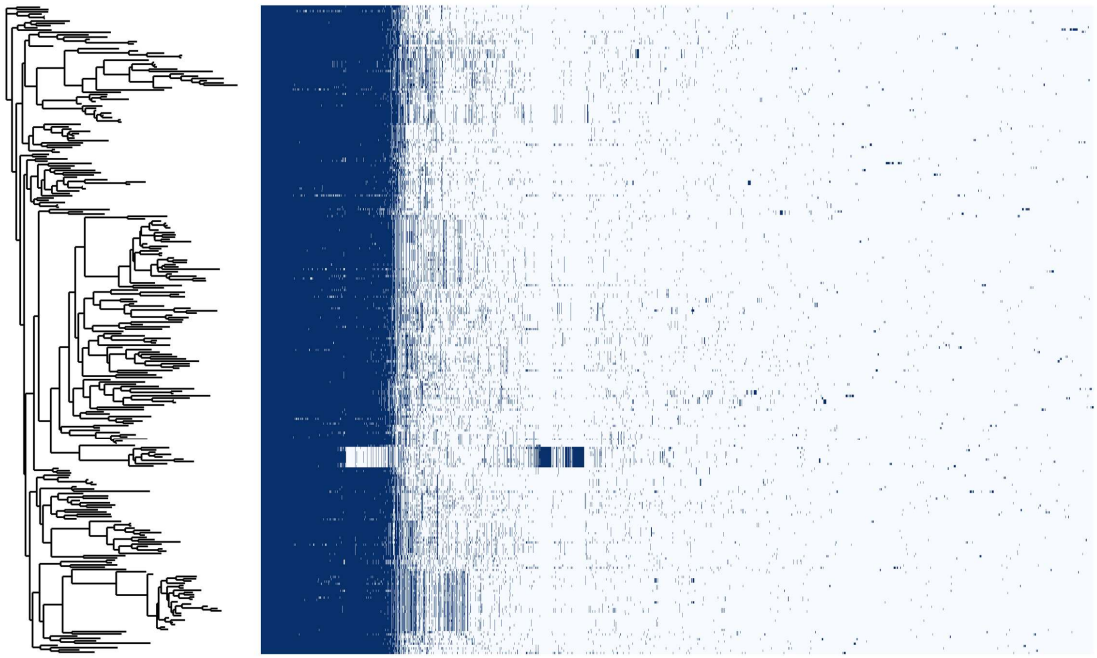


Figure 6: Phylogenetic tree of clinically relevant PA The tree is based on core genome alignment. Gene clustering is performed with 95% nucleotide identity over 95% of sequence length. Presence of gene cluster in a strain is shown as blue, and absence is shown as white.

3.2 Designing network architecture with 4-fold CV

Based on several combinations of hyperparameters choices, roughly 150 different architectures were tried. Only the top-performing model, which we henceforth call `PANet` (PAngenome Network, (**Fig. 7**)) is shown and discussed. For `PANet`, the first convolutional block learns three convolutional filters; hence three feature maps are obtained at the end of ConvBlock-1. For simplicity, the design of the convolutional blocks was kept constant, as shown in (**Fig. 8**). After each convolutional block, three additional filters are added in each block. After five such convolutional blocks, 15 feature maps (from the respective convolutional kernels) are obtained. These are flattened and stacked up then connected to the fully connected network (**Fig. 9**).

In the entire network, `PANet` has five convolutional blocks and one fully connected block. Hence there are a total of 17 hidden layers in the network, 5x(3 of each convolutional block) and two hidden layers of fully connected neurons. Each convolutional block comprises a total of three hidden layers, i.e. two convolution layers and one batch-normalization (BN) layer. The max-pooling layer is not counted as a hidden layer, as there are no trainable parameters.

Note that after the 1st ConvBlock-1 (**Fig. 7**), the effective number of neurons increases, before getting compressed in the further convolutional blocks. Upon adding another convolutional block lead to a reduction in validation-accuracy, most likely as the network was losing relevant information as one convolution block ends with a max-pooling layer. As the max-pooling reduces the length of feature map in half, the reduced number of neurons could not be compensated by learning additional three convolutional kernels that each CB (**Fig. 8**) adds to a network. Different structures of convolutional blocks were experimented with, e.g. number of filters to be added in each block, number of convolutional operations in each block, different strides, different kernel length, etc. This particular arrangement of adding three convolutional filters in each blocks (**Fig. 7**), performed the best.

To address statistical underpowering of the problem, as mentioned in **Sec. 1.2**, we employ the following regularization methods. First, we use a drop-rate of 25% in the fully connected block. Drop-out has been shown to prevent over-fitting significantly and outperforms other regularisation methods, such as involving L1 or L2 norms (Srivastava et al., 2014). Early stopping is another regularisation method that mitigates the over-fitting of DNNs.

3.3 Architecture of `PANet`

We find that `PANet` (**Fig. 7**) performed the best on the cross-validation procedure, amongst the roughly 150 architectures that were experimented in this project. Note that despite trying to minimize the model complexity (by tuning the computationally expensive fully-connected layer), more than 98% of the trainable parameters occur at the bottleneck, when the 15 feature maps are flatted, and connected to a 12-neuron (dense) layer (**Tab.2**).

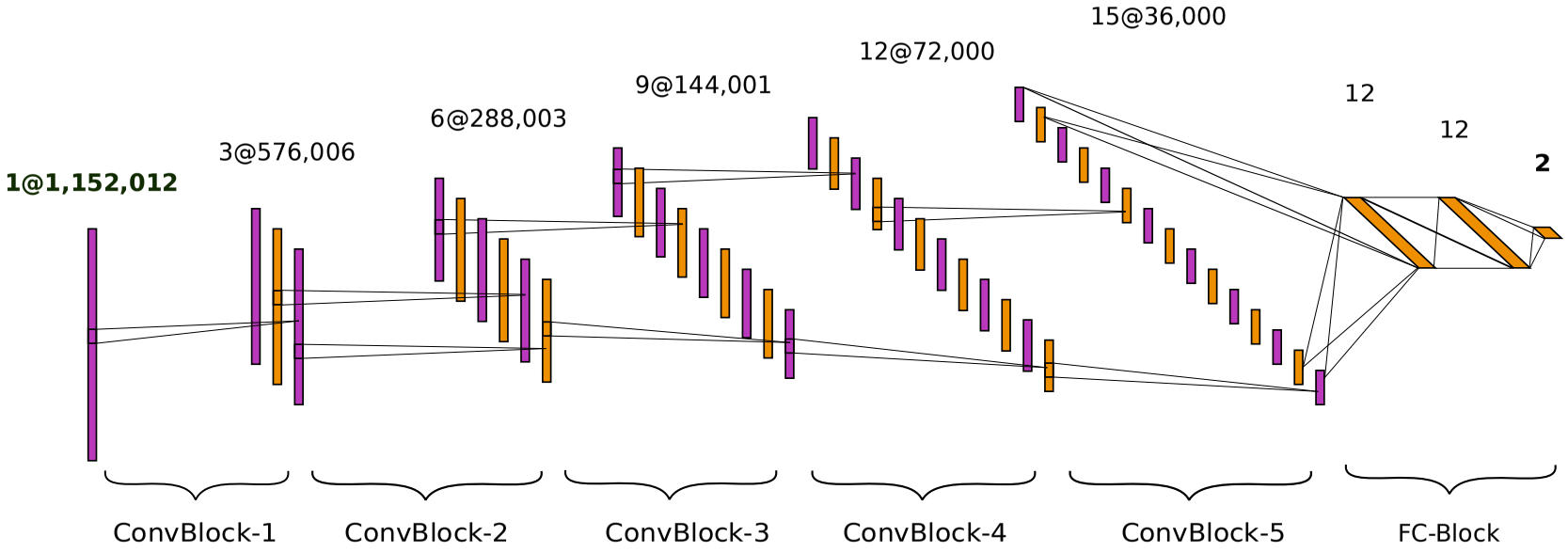


Figure 7: DNN architecture- The DNN consists of five convolutional blocks (**Fig. 9**), followed by a fully connected block (**Fig. 9**) that concatenates and flattens the incoming neurons. Each of the five representative feature map (one feature map is obtained after each convolutional block are shown as pink-orange array of neurons, arranged according to feature maps. For all hidden layers in the convolutional blocks, except the input layer, the number preceding the **@** symbol indicates the number of output channels after each convolutional block. The input and output neurons are mentioned in boldface. The number of parameters to be estimated are as indicated (**Tab. 2**)

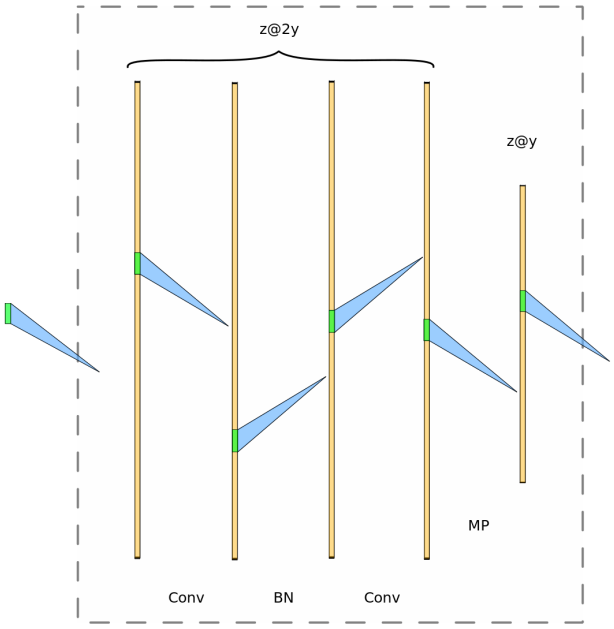


Figure 8: ConvBlock-x A single block in the network (**Fig. 7**) consists of a total of two convolution (Conv) layers, one batch-normalization (BN) layers, and one max-pooling (MP) layers. The number of channels in each ConvBlock-x differs, where $x \in 1, 2, 3, 4, 5$ has 3, 6, 9, 12, and 15 channels respectively.

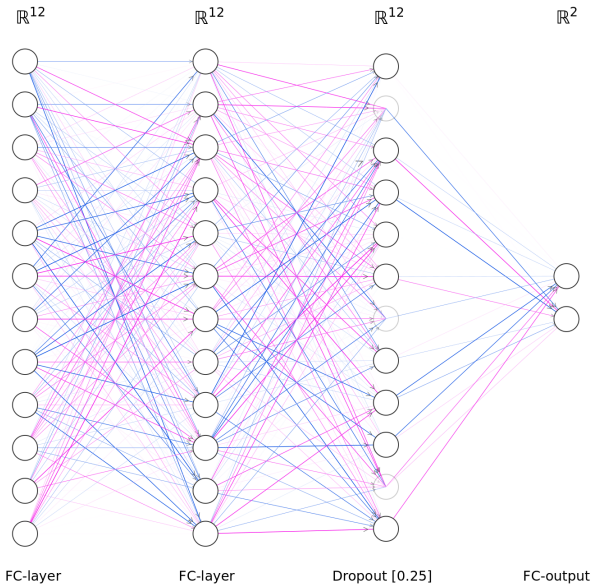


Figure 9: FC-Block After the ConvBlocks $(1-5)$, the 15 convolutional channels are first 'flattened'. The flattened neurons are then fed to the FC-Block, which consists of 2 fully connected layers (of 12 neurons each) that are followed by a 25% dropout layer, and precedes the binary output phenotype. The dropout layer introduces sparsity in the network and simultaneously prevents neuronal co-adaptation (Srivastava et al., 2014; Hinton et al., 2012).

Table 2: PANet model complexity Parameters of DNNs can help estimate its model complexity. PANet comprises a total of 6,501,914 parameters, of which an overwhelming majority (98.44% or 6,480,012) are in the fully connected block (f). Huge computational demand posed by fully connected neurons was the motivation behind employing convolution kernels (**Fig. 8**) that ultimately led to the design of the novel PANet architecture (**Fig. 7**).

(a) ConvBlock-1

Layer	Output dimensions (length, channels)	Parameters
Convolution	(1152012, 3)	78
BatchNormalization	(1152012, 3)	12
Convolution	(1152012, 3)	228
MaxPooling	(576006, 3)	0

(b) ConvBlock-2

Layer	Output dimensions (length, channels)	Parameters
Convolution	(576006, 6)	456
BatchNormalization	(576006, 6)	24
Convolution	(576006, 6)	906
MaxPooling	(288003, 6)	0

(c) ConvBlock-3

Layer	Output dimensions (length, channels)	Parameters
Convolution	(288003, 9)	1359
BatchNormalization	(288003, 9)	36
Convolution	(288003, 9)	2034
MaxPooling	(144001, 9)	0

(d) ConvBlock-4

Layer	Output dimensions (length, channels)	Parameters
Convolution	(144001, 12)	2712
BatchNormalization	(144001, 12)	48
Convolution	(144001, 12)	3612
MaxPooling	(72000, 12)	0

(e) ConvBlock-5

Layer	Output dimensions (length, channels)	Parameters
Convolution	(72000, 15)	4515
BatchNormalization	(72000, 15)	60
Convolution	(72000, 15)	5640
MaxPooling	(36000, 15)	0

(f) FC-Block

Layer	Output dimensions (no. of neurons)	Parameters
Flatten	(540000)	0
Dense	(12)	6480012
Dense	(12)	156
Dropout	(12)	0
Dense	(2)	26

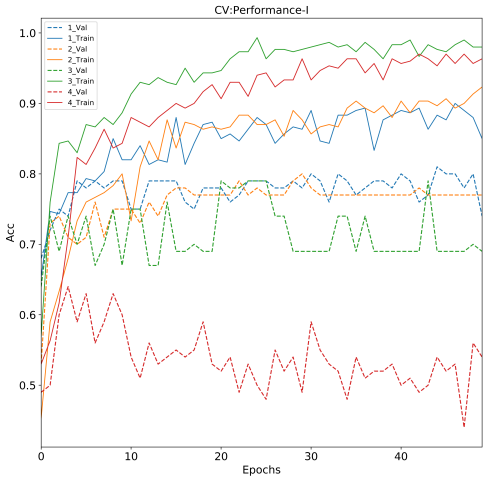
3.4 Number of final-training epochs: employing early-stopping

After fixing the final architecture of PAnet, next, the appropriate number of training iterations need to decided. An epoch is one round of iteration where a complete pass through all the training examples takes place. During model training, stochastic gradient descent computes gradients over mini-batches of the training-data, and not the entire training-data. Batch-size of 10 was employed; hence there are 20 minibatches in each epoch of model training.

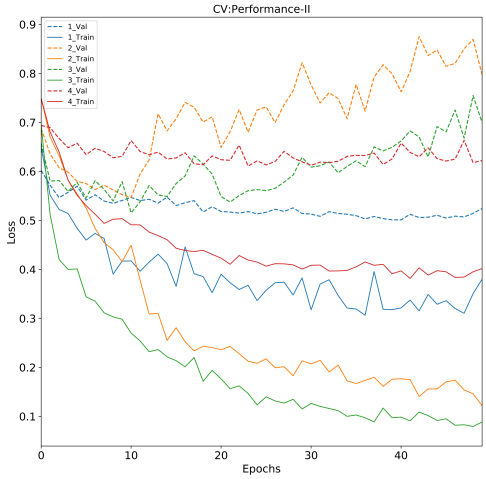
Typically line plots known as learning curves can assist in visualizing model training. Learning curves are obtained by plotting epochs along the x-axis and the training accuracy or loss function on the y-axis. These help in diagnosing whether a model has under-learnt, over-learnt (over-fitting), or is suitably fit to the training-dataset. We ran cross-validation for 50 epochs as an example of model training visualization.

From the learning curves (**Fig. 10**), it can be observed that somewhere between 10-20 epochs, the model begins to saturate, thus risking over-fitting. Employing early-stopping regularisation

(with 'patience parameter- 7'), suggested that performing model-training for 11 epochs was a viable option, and suitably curtails overfitting. For image recognition tasks, epochs may be of the order of hundreds or thousands, but as our panel is tiny in comparison to ImageNet datasets, PANet's performance seems reasonable.



(a) Accuracy

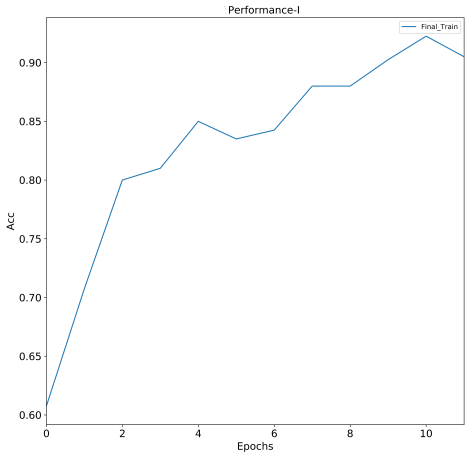


(b) Loss

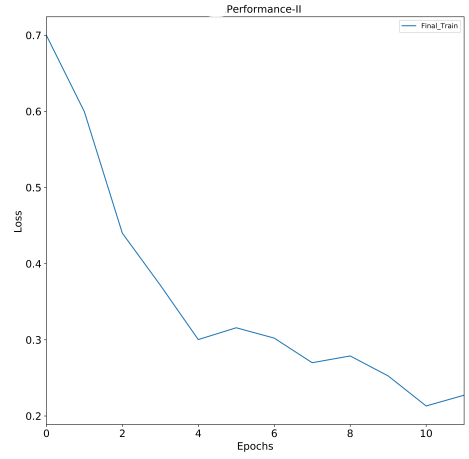
Figure 10: Hyperparameter tuning via 4-fold CV A typical cross-validation example that was trained for 50 epochs. Model over-fitting is clearly seen somewhere between 10-15 epochs. Particularly after 15 epochs, the validation-loss increases while no gain in validation-accuracy is observed.

3.5 Final model training

Having fixed the architecture and selected the appropriate hyper-parameters, we train the network on the entire training-dataset. Early stopping based on validation-loss was employed to curtail over-fitting. Patience parameter- 7 was used during the final model training(**Fig. 11**). The learning curves for the final training of the PANet model show the possible limit on training data. The increase in training accuracy is not strictly monotonic, with few instances wherein the training accuracy decreases (e.g. 5th, 8th, and 11th epoch) (**Fig. 11a**). This behaviour is expected as we use a stochastic adaptive solver. Similarly, the decreasing binary cross-entropy loss trend seems to be inversely correlated of the training accuracy (**Fig. 11b**). The **Fig. 11** serves as a sanity-check and must be ensured before final model training.



(a) Accuracy



(b) Loss

Figure 11: Training PANet on entire training-data Since early-stopping is employed, it can be observed that the model could theoretically make predictions that are upto 90% accuracy. This model suitably avoids over-fitting. Note that fully learning (i.e. upto 100% validation-accuracy) the training data is a highly trivial exercise.

3.6 Model testing and performance estimates

For estimating the unbiased performance of binary classifiers, several metrics are used in ML literature. Precision, Recall, and Accuracy informs about the trade-offs in the classification of one class against the other for a classifier, while AUROC, F1, and MCC estimate performance across different classifiers. Precision is defined as the fraction of correct classification of resistant class, overall microbes that are predicted as resistant. Recall is calculated as the correct prediction rate of the resistant class. Accuracy is the fraction of total correct classification (both classes), overall predictions. F1 score is the harmonic mean of sensitivity and precision. Matthews correlation coefficient or MCC is calculated (**Eq. 4**), with TP, TN, FP, FN being true positive, true negative, false positive, and false negative respectively.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{4}$$

For the testing, we performed both balanced and imbalanced testing. For creating a balanced test set, a 5-fold up-sampling of the minority class (7 resistant strains) was performed to maintain equal abundance as the majority class (35 sensitive strains). As the `PANet` model can learn from only 40 training examples of resistant *P. aeruginosa*, its ability to generalize for resistant strains is more complicated, as compared to the prediction of sensitive (trained on 200 examples) strains. The lower Precision score for *Imbalanced* testing-data reflects the difficulty.

The ROC curves for both the imbalanced and balanced testing are the same, but different performance metrics are obtained. Note that the default threshold cutoff of 0.5, gives the following performance estimates (**Tab. 3**). Thresholds are typically based on user preference, on how they want to balance trade-offs. Typically, the threshold is based on user preference and can be balanced on the desired level of trade-off between Precision and Recall.

All the metrics lie between 0 and 1, wherein the values closer to 1 are better. Performance metrics (**Tab. 3**) and confusion matrices (**Tab. 4**) for our test data are shown.

Table 3: Performance metric estimates The *Balanced* testing-data inflates metrics such as precision, F1 and MCC, while the AUROC and recall remain the same for both testings. While *Imbalanced* testing-data estimates higher accuracy, it reports smaller precision estimate than *Balanced* testing-data.

	AUROC	**Accuracy**	**Precision**	**Recall**	**F1**	**MCC**
Balanced	0.698	0.7	0.769	0.571	0.656	0.414
Imbalanced	0.698	0.786	0.4	0.571	0.471	0.366

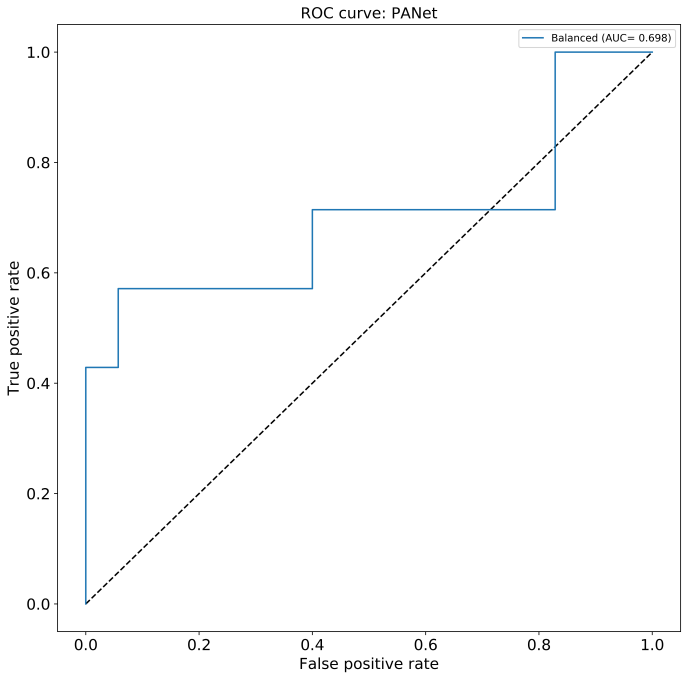


Figure 12: PANet ROC curve - For the most part, the PANet performs much better than random guess (y=x line). Particularly by employing a higher confidence threshold, and remaining in the upper right half of the ROC plot, good classification can be obtained.

Table 4: PANet **Confusion matrices** As only the true resistant class vary between balanced and imbalanced testing, the impact of 5-fold oversampling of the resistant class can be clearly observed in the second column of both the confusion matrices.

(a) Balanced testing

	True sensitive	**True resistant**
Predicted sensitive	29	15
Predicted resistant	6	20

(b) Imbalanced testing

	True sensitive	**True resistant**
Predicted sensitive	29	3
Predicted resistant	6	4

Also note that the thresholds have not been indicated for the ROCs. All the three results, namely, the ROC curve (**Fig. 12**), performance metrics (**Tab. 3**), and confusion matrix (**Tab. 4**) when taken together guide the user in estimating true classifier performance on a real world data.

3.7 Comparison to elastic net

Intriguingly, both elastic net linear regression (**Fig. 13**) and `PANet` (**Fig. 12**) have identical AUROC. The shape of both ROCs differ slightly, but otherwise, the two curves are almost identical. Together, it suggests that `PANet` might be modelling a close approximate to an LR, perhaps even resembling an elastic net.

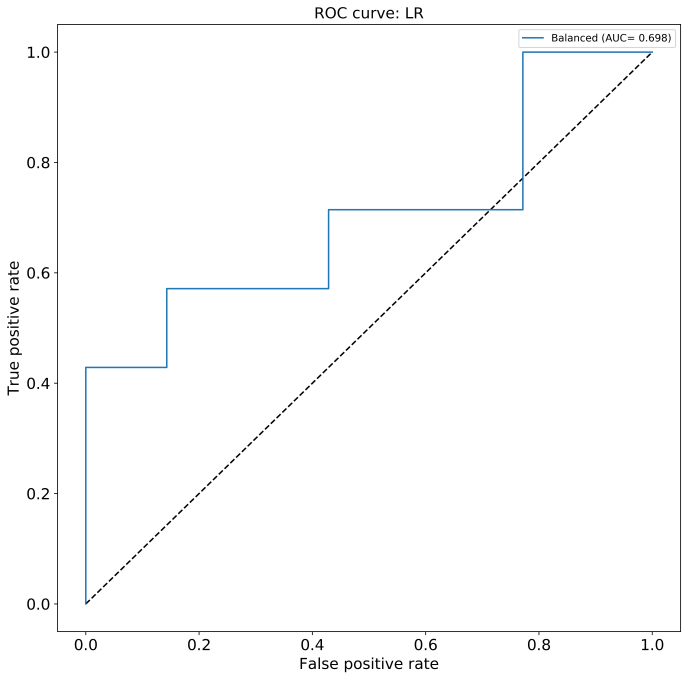


Figure 13: Elastic net ROC curve The ROC curve is LR is similar to its `PANet` counterpart. The confusion matrix for the elastic net (not shown) is exactly the same as `PANet`, for both the balanced and imbalanced testing.

While the coefficients of the elastic net regression model have not been thoroughly investigated, it might be conceivable that the elastic net could capture variants with large effect size. Such a coarse approximation may be explaining most of the variation in the unitig presence-absence pattern, that is essential for class separation. However, variants with small coefficients (hence small effect sizes), would be dropped because of the sparse nature of the L1 norm. To this end, DNNs might be able to unravel more molecular markers. Indeed, a confirmatory experiment would be unbiased comparison of important unitigs (salient features) of both LR and `PANet` models.

4 Discussion

We made comparisons to LR with elastic net regularisation, to quantify improvements that DNNs provide over linear methods, and indeed if the combination of variants can explain the data better than traditional models. Despite our best efforts, for the current panel of it seems that DNNs can only perform as well as linear models, and no improvements in classification performance were observed. Such a result may be explainable if some certain, possibly small, set of unitigs can broadly describe the phenotype.

We employed 25 as the length of convolutional filters, and experimented with other options of increasing filter length, changing convolutional strides, or changing the number of the fully connected neurons etc. No particularly consistent or striking pattern could be observed from experiments during hyperparameter tuning. A lack of trend in hyperparameter tuning is typical challenges in deep learning that impede the choice of appropriate network architectures. Except for the fact that changes were only accepted if it increased the cross-validation accuracy, the tuning process appeared random, and decisions were purely based on heuristics. As an illustration, in the final `PANet` model, the network had a bottleneck of 12 neurons, in the FC-block (**Fig. 9**), which drastically reduced the model complexity (total number of trainable parameters such as weights and biases to be estimated). While compressing 540,000 neurons (from 15 feature maps, each of length 36000 neurons) to 12 neurons seems like an abrupt jump; still, this choice had the best validation-accuracy. Of note, the final network had been reduced in size to the extent that the standard (algorithmic) optimization implementations were not able to train networks (under default settings). Smaller network architectures are desirable as they are less prone to over-fitting. Heuristic recommendation favours smaller networks, as they limit unnecessary computational complexity and reduce overfitting possibilities.

Computational and statistical challenges

Statistical association testing in GWAS and supervised machine learning are fundamentally different approaches to the genotype-phenotype mapping problem. The former is concerned with finding individual variations that are strongly associated with the phenotype of interest. The latter, as we show in our deep learning approach, tries to find combinations of variations and their interaction that aid class separation.

For microbial genomes, the total genomic variations are often many orders of magnitude greater than the panel size. Hence, high genomic plasticity severely impacts statistical power in association testing, allowing us to uncover only common variants associated with resistance. From human GWAS, alleles with minor allele frequency (MAF) more than 5% are termed as common variants (Romagnoni et al., 2019). This definition is currently unknown for microbial organisms and would extensively vary across species. Since microbes are much more genetically diverse, it would be interesting to quantify the extent to which DNNs or linear models could assist in uncovering variants association, on the 'common versus rare' continuum. For instance, (Romagnoni et al., 2019) also analyzed the MAF limit and corresponding AUROCs.

Furthermore, the results in this project might be restricted by the small cystic fibrosis *P. aeruginosa* panel. As DNNs generally perform well on large-scale training-data, a small panel limits approximation capacity of the DNNs. We nonetheless show that deep learning is indeed possible on small biological data sets, further opening up avenues for exploring DNNs in the con-

text of genotype-phenotype mapping. Nevertheless, including domain knowledge and expertise might be able to bolster the approximation power of DNNs.

4.1 Relevant deep learning literature

Interestingly, similar results were recently published by (Bellot et al., 2018), where they reported that 'CNN performance was competitive to linear models', and that they 'did not find any case where DL outperformed the linear model by a sizeable margin'. Albeit the Bellot et al. (2018) study was on human genetic data, their sample size was much larger - consisting of ∼500k single nucleotide polymorphisms (SNPs) from ∼100k individuals. It has been theorized that higher eukaryotes embody greater epistasis in their genomes, as compared to prokaryotes. However, despite capturing potential non-linearities, the DNNs did not augment classification performance.

Similarly, a recent paper (Romagnoni et al., 2019) compared the performance of machine learning methods for classification of Crohn's disease based on genotype data. In their work, machine learning methods were able to detect all variants previously identified by GWAS as the best predictors of disease (Romagnoni et al., 2019) but also discovered additional predictors with smaller effect-sizes.

4.2 Outlook

Several intriguing follow-up experiments are posed by our current work:

1. Corrections for population structure could be incorporated by simultaneously training two (deep) neural networks, one for the presence-absence of unitigs and the other for incorporating covariance (with, e.g. population structure corrections) - analogous to the LMM 1, but with adaptive non-linearity of DNNs.

2. Greedy solutions to the NP-hard problem of graph clustering could be a boon to future investigations on genotype-phenotype mapping. A biologically meaningful partial ordering that imposes particular spatial structure within a graph pangenome might augment performance. Given that we use filters of length 25, it could be worthwhile to cluster unitigs in spatial range of 25 nodes, before training of the models. Such clustering or sorting should be experimented with, to see if it can capture local structures among unitigs. We tried to sort the raw output of the compacted de Bruijn Graph into a directed acyclic graph (one without closed loops in graph traversal), but it led to irresolvable conflicts within the graph.

3. From DNNs, 'scores' that describe statistical importance of input nodes can be obtained, e.g. through generating saliency maps (Simonyan et al., 2014; Pan et al., 2016). By computing the gradient with respect to input neurons in a fully trained network, the important features (unitigs) associated with classification may be revealed. Nevertheless, despite obtaining such scores (as proxies of statistical confidence), they, unfortunately, cannot be directly compared to (contemporary) standard statistics such as p-values.

4. An interesting and relevant future work would be to include biologically relevant properties of each genome. As a follow-up hypothesis, we speculate that increasing the phenotypic

complexity of the network may aid in improving genotype-phenotype mapping. For instance, most of the eco-evolutionary works on microbial evolution investigate antibiotics-like genes in tandem with toxicity-like genes (van Dijk and Hogeweg, 2016). In natural ecologies, antibiotic resistance and genes for toxic metabolites are used as strategies of resources competition. Carrying additional accessory genes such as resistance (or toxicity) genes can be expensive, especially if no strain carries toxicity genes (van Dijk and Hogeweg, 2016). Without a selection pressure, genes that confer no immediate fitness benefit (in the absence of toxicity genes), would be lost or be outcompeted by another strain with higher fitness. Analogously, the loss function for a complex phenotype (e.g. ability to form biofilms, toxicity genes, mobile genetic elements, etc.; where N is the total number of classes) can be easily extended in neural networks (**Eq. 5**).

$$Cross - entropyloss = -\frac{1}{N}\sum_{i=1}^{N} q_i \cdot \log\left(p_i\right) + \left(1 - q_i\right) \cdot \log\left(1 - p_i\right) \tag{5}$$

5. There might be a theoretical limit to which one can classify cases against control, purely based on presence-absence of polymorphisms. While we do not claim that our results (linear models or equivalent DNNs that we have tested in this project) are the limit, a limit must indeed exist - as unitig sequences themselves are not used. Only unitigs presence-absence were employed in the mapping. A hard limit might be 75%, 80%, or even 85% - and currently remains to be uncovered for *P. aeruginosa*, and could be specific for individual panels.

Currently, extraction of interpretable knowledge from DNN, remains an unsolved problem in deep learning (Najafabadi et al., 2015; Bengio et al., 2013), especially given that any potential non-linear interactions between individual variants, will be compounded into increasingly abstract representations in the deeper hidden layers of neural networks. Pertinent to genotype-phenotype mapping, gaining intuitions underlying how DNNs make decisions would ultimately lead to a greater mechanistic understanding of the genetics of pangenomes, that are associated with phenotypes. Tackling this problem in microbial genomics has immense application in managing infectious diseases; revealing precise features of microbial evolution; is of medical & industrial interest and has importance in eco-evolutionary research. For further challenges along these lines in pattern recognition, biological intuitions could perhaps serve as inspirations for future research.

5 Conclusions

Towards building scalable machine learning methods that can accommodate increasing quantities of high-dimensional genomics data, this project was aimed to serve as a proof of concept for applications of deep learning in (microbial) genomics tailored towards genotype-phenotype mapping. Before this current thesis, no previously published network architecture was publicly available. We are the first to demonstrate the application of `PANet`, a novel DNN for mapping antimicrobial resistance of a highly plastic *P. aeruginosa* pangenome. The `PANet` architecture may be suitably adapted for other prokaryotic or eukaryotic genomes, and domain knowledge could be included. We highly encourage rigorous experimentation with other hyperparameter choices that govern DNN architectures.

We find that despite experimenting with several network architectures, the DNNs are as competitive as linear models. The results are exemplified by the fact that the ROC curves both the DNN and linear model surprisingly similar; with the AUROCs also being identical. These results might suggest either of two scenarios: given the limitations imposed by small panel size, a linear model may be the best explanation that maps the variations described by unitigs of the pangenome. Alternatively, the results could imply that antimicrobial resistance genotype-phenotype mapping might indeed have a linear relationship.

These results suggest that further research is needed to suitably adopt convolutional neural networks to tackle computational challenges in pangenomics and may soon be able to outperform linear models. Gaining a further mechanistic understanding of the neural network decision making will aid in translating better therapeutics for infectious diseases and will ultimately assist in strategies of global epidemiology.

6 Layman's summary

The Deoxyribonucleic acid (DNA) of every living organism on planet earth; from microorganisms to insects, to vertebrates, to primates, comprises of four nucleotides or base: adenine (A), cytosine (C), guanine (G), and thymine (T). Gaining the ability to thoroughly understand and interpret the sequences of bases, hidden from plain sight within the organism DNA, has been a long-standing open problem in biology. While the methods based on sequence similarity (from shared common ancestry), are the workhorse for identification of well-described genetic fragments associated with a phenotype, how are new genetic markers (variations such as mutations in genes) discovered? Genome-wide association studies (GWAS) are such efforts that seek to unravel the genetics underlying given phenotypes (observable properties). Historically, GWAS have involved methods assuming a linear relationship between genotype (genomic description of all variation between organisms) and phenotype of interest (such as resistance or susceptibility towards antimicrobial drugs).

For several cases, the assumptions of linearity may not necessarily hold, thus creating an impetus for the development of novel methods that can accommodate the real complexity of genomic datasets. At the same time, DNNs in this past decade has vastly accelerated our ability to find patterns and automate tasks. However, their application in genomics awaits scientific rigour. The primary goal of the work in this thesis is to assess the feasibility, practicality, and gain in performance when employing DNNs for genotype-phenotype mapping - as compared to traditional linear methods.

We employ a method published by Jaillard et al. (2018) for describing the genomes of *Pseudomonas aeruginosa* that are resistant to a particular antimicrobial (Amikacin) and experiment with different configurations of neural networks for achieving excellent genotype-phenotype mapping performance. Upon extensive experimentation, we find that none of the architectures could outperform a particular variation of a linear method (namely elastic net linear regression). Our results challenge other works in genomics, that claim that deep learning is far superior to traditional methods, and these must be taken with fair criticism. Deep learning is not a panacea after all, and further research (**Sec. 4.2**) is needed to appropriately adapt DNNs that were initially motivated by image analysis, to soon be competitive or even outperform linear methods in future.

References

Alexander Amini. Introduction to Deep Learning, 2020. URL http://introtodeeplearning.com/slides/6S191{_}MIT{_}DeepLearning{_}L1.pdfintrotodeeplearning.com.

Pau Bellot, Gustavo de los Campos, and Miguel Pérez-Enciso. Can deep learning improve genomic prediction of complex human traits? *Genetics*, 210(3):809–819, nov 2018. ISSN 19432631. doi: 10.1534/genetics.118.301298.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2013. ISSN 01628828. doi: 10.1109/TPAMI.2013.50.

Caitlin Collins and Xavier Didelot. A phylogenetic method to perform genome-wide association studies in microbes that accounts for population structure and recombination. *PLoS Comput. Biol.*, 14(2), feb 2018. ISSN 15537358. doi: 10.1371/journal.pcbi.1005958.

Otto X. Cordero and Martin F. Polz. Explaining microbial genomic diversity in light of evolutionary ecology, 2014. ISSN 17401534.

Purushottam D. Dixit, Tin Yau Pang, and Sergei Maslov. Recombination-driven genome evolution and stability of bacterial species. *Genetics*, 207(1):281–295, 2017. ISSN 19432631. doi: 10.1534/genetics.117.300061.

Yohei Doi and Yoshichika Arakawa. 16S Ribosomal RNA Methylation: Emerging Resistance Mechanism against Aminoglycosides. *Antimicrob. Resist. Clin. Infect. Dis.*, 45:88–94, 2007. doi: 10.1086/518605. URL https://academic.oup.com/cid/article-abstract/45/1/88/481913.

W. Michael Dunne, Magali Jaillard, Olivier Rochas, and Alex Van Belkum. Microbial genomics and antimicrobial susceptibility testing, 2017. ISSN 17448352.

Sarah G. Earle, Chieh Hsi Wu, Jane Charlesworth, Nicole Stoesser, N. Claire Gordon, Timothy M. Walker, Chris C.A. Spencer, Zamin Iqbal, David A. Clifton, Katie L. Hopkins, Neil Woodford, E. Grace Smith, Nazir Ismail, Martin J. Llewelyn, Tim E. Peto, Derrick W. Crook, Gil McVean, A. Sarah Walker, and Daniel J. Wilson. Identifying lineage effects when controlling for population structure improves power in bacterial association studies. *Nat. Microbiol.*, 1(5), apr 2016. ISSN 20585276. doi: 10.1038/nmicrobiol.2016.41.

Antonino Fiannaca, Laura La Paglia, Massimo La Rosa, Giosue' Lo Bosco, Giovanni Renda, Riccardo Rizzo, Salvatore Gaglio, and Alfonso Urso. Deep learning models for bacteria taxonomic classification of metagenomic data. *BMC Bioinformatics*, 2018. ISSN 14712105. doi: 10.1186/s12859-018-2182-6.

Erik Garrison, Jouni Sirén, Adam M. Novak, Glenn Hickey, Jordan M. Eizenga, Eric T. Dawson, William Jones, Shilpa Garg, Charles Markello, Michael F. Lin, Benedict Paten, and Richard Durbin. Variation graph toolkit improves read mapping by representing genetic variation in the reference, oct 2018. ISSN 15461696.

Boris Hanin and Mark Sellke. Approximating continuous functions by relu nets of minimal width. *arXiv*, oct 2017. URL http://arxiv.org/abs/1710.11278.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, volume 2016-Decem, pages 770–778. IEEE Computer Society, dec 2016. ISBN 9781467388504. doi: 10.1109/CVPR.2016.90. URL http://image-net.org/challenges/LSVRC/2015/.

G E Hinton, N Srivastava, A Krizhevsky, I Sutskever, and R R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv*, 2012.

Gabriel E. Hoffman. Correcting for Population Structure and Kinship Using the Linear Mixed Model: Theory and Extensions. *PLoS One*, 8(10), oct 2013. ISSN 19326203. doi: 10.1371/journal.pone.0075707.

Kurt Hornik. Approximation Capabilities of Multilayer Feedforward Networks. *Neural Networks*, 4:251–257, 1991. doi: 10.1016/0893-6080(91)90009-T.

Magali Jaillard, Alex van Belkum, Kyle C. Cady, David Creely, Dee Shortridge, Bernadette Blanc, E. Magda Barbu, W. Michael Dunne, Gilles Zambardi, Mark Enright, Nathalie Mugnier, Christophe Le Priol, Stéphane Schicklin, Ghislaine Guigon, and Jean Baptiste Veyrieras. Correlation between phenotypic antibiotic susceptibility and the resistome in Pseudomonas aeruginosa. *Int. J. Antimicrob. Agents*, 50(2):210–218, 2017. ISSN 18727913. doi: 10.1016/j.ijantimicag.2017.02.026.

Magali Jaillard, Leandro Lima, Maud Tournoud, Pierre Mahé, Alex van Belkum, Vincent Lacroix, and Laurent Jacob. A fast and agnostic method for bacterial genome-wide association studies: Bridging the gap between k-mers and genetic events. *PLoS Genet.*, 2018. ISSN 1553-7404. doi: 10.1371/journal.pgen.1007758.

Gunther Jansen, Niels Mahrt, Leif Tueffers, Camilo Barbosa, Malte Harjes, Gernot Adolph, Anette Friedrichs, Annegret Krenz-Weinreich, Philip Rosenstiel, and Hinrich Schulenburg. Association between clinical antibiotic resistance and susceptibility of Pseudomonas in the cystic fibrosis lung. *Evol. Med. Public Heal.*, 2016(1):182–194, 2016. ISSN 20506201. doi: 10.1093/EMPH/EOW016.

Moshe Leshno, Vladimir YA. Lin, Allan Pinkus, and Shimon Schoken. Multilayer Feedforward Networks With a Nonpolynomial Activation Function Can Approximate Any Function. *Neural Networks*, 6(6):861–867, 1993. ISSN 08936080. doi: 10.1016/S0893-6080(05)80131-5.

Mingzhi Lin and Edo Kussell. Inferring bacterial recombination rates from large-scale sequencing datasets. *Nat. Methods*, 16(2):199–204, feb 2019. ISSN 15487105. doi: 10.1038/s41592-018-0293-7.

Zhou Lu, Hongming Pu, Feicheng Wang, Zhiqiang Hu, and Liwei Wang. The Expressive Power of Neural Networks: A View from the Width. *Adv. Neural Inf. Process. Syst.*, 2017-Decem: 6232–6240, 2017. ISSN 10495258.

Trudy F.C. Mackay. Epistasis and quantitative traits: Using model organisms to study gene-gene interactions, 2014. ISSN 14710056.

Tobias Marschall, Manja Marz, Thomas Abeel, Louis Dijkstra, Bas E Dutilh, Ali Ghaffaari, Paul Kersey, Wigard P Kloosterman, Veli Mäkinen, Adam M Novak, Benedict Paten, David Porubsky, Eric Rivals, Can Alkan, Jasmijn A Baaijens, Paul I W De Bakker, Valentina Boeva, Raoul J P Bonnal, Francesca Chiaromonte, Rayan Chikhi, Francesca D Ciccarelli, Robin Cijvat, Erwin Datema, Cornelia M Van Duijn, Evan E Eichler, Corinna Ernst, Eleazar Eskin, Erik Garrison, Mohammed El-Kebir, Gunnar W Klau, Jan O Korbel, Eric-Wubbo Lameijer, Benjamin Langmead, Marcel Martin, Paul Medvedev, John C Mu, Pieter Neerincx, Klaasjan Ouwens, Pierre Peterlongo, Nadia Pisanti, Sven Rahmann, Ben Raphael, Knut Reinert, Dick de Ridder, Jeroen de Ridder, Matthias Schlesner, Ole Schulz-Trieglaff, Ashley D Sanders, Siavash Sheikhizadeh, Carl Shneider, Sandra Smit, Daniel Valenzuela, Jiayin Wang, Lodewyk Wessels, Ying Zhang, Victor Guryev, Fabio Vandin, Kai Ye, and Alexander Schönhuth. Computational pan-genomics: status, promises and challenges. *Brief. Bioinform.*, 19(1):bbw089, oct 2016. ISSN 1467-5463. doi: 10.1093/bib/bbw089. URL https://academic.oup.com/bib/article-lookup/doi/10.1093/bib/bbw089.

Maryam M. Najafabadi, Flavio Villanustre, Taghi M. Khoshgoftaar, Naeem Seliya, Randall Wald, and Edin Muharemagic. Deep learning applications and challenges in big data analytics. *J. Big Data*, 2015. ISSN 21961115. doi: 10.1186/s40537-014-0007-7.

Andrew J. Page, Carla A. Cummins, Martin Hunt, Vanessa K. Wong, Sandra Reuter, Matthew T.G. Holden, Maria Fookes, Daniel Falush, Jacqueline A. Keane, and Julian Parkhill. Roary: Rapid large-scale prokaryote pan genome analysis. *Bioinformatics*, 2015. ISSN 14602059. doi: 10.1093/bioinformatics/btv421.

Junting Pan, Elisa Sayrol, Xavier Giro-I-Nieto, Kevin Mcguinness, and Noel E. O'connor. Shallow and Deep Convolutional Networks for Saliency Prediction. In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, volume 2016-Decem, pages 598–606. IEEE Computer Society, dec 2016. ISBN 9781467388504. doi: 10.1109/CVPR.2016.71. URL https://github.com/.

Gabriele Pedruzzi, Ayuna Barlukova, and Igor M. Rouzine. Evolutionary footprint of epistasis. *PLoS Comput. Biol.*, 2018. ISSN 15537358. doi: 10.1371/journal.pcbi.1006426.

Miguel Pérez-Enciso and Laura M. Zingaretti. A guide for using deep learning for complex trait genomic prediction, jul 2019. ISSN 20734425.

Gerald Pier. Application of vaccine technology to prevention of Pseudomonas aeruginosa infections, oct 2005. ISSN 14760584.

Divyae Kishore Prasad. PANet: PAngenome Network, 2019. URL https://github.com/divprasad/1DconvKernels.

Alberto Romagnoni, Simon Jégou, Kristel Van Steen, Gilles Wainrib, Jean Pierre Hugot, Laurent Peyrin-Biroulet, Mathias Chamaillard, Jean Frederick Colombel, Mario Cottone, Mauro D'Amato, Renata D'Incà, Jonas Halfvarson, Paul Henderson, Amir Karban, Nicholas A. Kennedy, Mohammed Azam Khan, Marc Lémann, Arie Levine, Dunecan Co Massey, Monica Milla, Sok Meng Evelyn Ng, Ioannis Oikonomou, Harald Peeters, Deborah D. Proctor, Jean Francois Rahier, Paul Rutgeerts, Frank Seibold, Laura Stronati, Kirstin M. Taylor, Leif Törkvist, Kullak Ublick, Johan Van Limbergen, André Andre Van Gossum, Morten H. Vatn, Hu Zhang, Wei Zhang, Jane M. Andrews, Peter A. Bampton, Murray Barclay, Timothy H. Florin, Richard Gearry, Krupa Krishnaprasad, Ian C. Lawrance, Gillian Mahy, Grant W. Montgomery, Graham Radford-Smith, Rebecca L. Roberts, Lisa A. Simms, Katherine Hanigan, Anthony Croft, Leila Amininijad, Isabelle Cleynen, Olivier Dewit, Denis Franchimont, Michel Georges, Debby Laukens, Harald Peeters, Jean Francois Rahier, Paul Rutgeerts, Emilie Theatre, André Andre Van Gossum, Severine Vermeire, Guy Aumais, Leonard Baidoo, Arthur M. Barrie, Karen Beck, Edmond Jean Bernard, David G. Binion, Alain Bitton, Steve R. Brant, Judy H. Cho, Albert Cohen, Kenneth Croitoru, Mark J. Daly, Lisa W. Datta, Colette Deslandres, Richard H. Duerr, Debra Dutridge, John Ferguson, Joann Fultz, Philippe Goyette, Gordon R. Greenberg, Talin Haritunians, Gilles Jobin, Seymour Katz, Raymond G. Lahaie, Dermot P. McGovern, Linda Nelson, Sok Meng Evelyn Ng, Kaida Ning, Ioannis Oikonomou, Pierre Paré, Deborah D. Proctor, Miguel D. Regueiro, John D. Rioux, Elizabeth Ruggiero, L. Philip Schumm, Marc Schwartz, Richard Regan Scott, Yashoda Sharma, Mark S. Silverberg, Denise Spears, A. Hillary Steinhart, Joanne M. Stempak, Jason M. Swoger, Constantina Tsagarelis, Wei Zhang, Clarence Zhang, Hongyu Zhao, Jan Aerts, Tariq Ahmad, Hazel Arbury, Anthony Attwood, Adam Auton, Stephen G. Ball, Anthony J. Balmforth, Chris Barnes, Jeffrey C. Barrett, Inês Barroso, Anne Barton, Amanda J. Bennett, Sanjeev Bhaskar, Katarzyna Blaszczyk, John Bowes, Oliver J. Brand, Peter S. Braund, Francesca Bredin, Gerome Breen, Matthew A. Morris J. Brown, Ian N. Bruce, Jaswinder Bull, Oliver S. Burren, John Burton, Jake Byrnes, Sian Caesar, Niall Cardin, Chris M. Clee, Alison J. Coffey, John MC Connell, Donald F. Conrad, Jason D. Cooper, Anna F. Dominiczak, Kate Downes, Hazel E. Drummond, Darshna Dudakia, Andrew Dunham, Bernadette Ebbs, Diana Eccles, Sarah Edkins, Cathryn Edwards, Anna Elliot, Paul Emery, David M. Evans, Gareth Evans, Steve Eyre, Anne Farmer, I. Nicol Ferrier, Edward Flynn, Alistair Forbes, Liz Forty, Jayne A. Franklyn, Timothy M. Frayling, Rachel M. Freathy, Eleni Giannoulatou, Polly Gibbs, Paul Gilbert, Katherine Gordon-Smith, Emma Gray, Elaine Green, Chris J. Groves, Detelina Grozeva, Rhian Gwilliam, Alistair S. Anita Hall, Naomi Hammond, Matt Hardy,

Pile Harrison, Neelam Hassanali, Husam Hebaishi, Sarah Hines, Anne Hinks, Graham A. Hitman, Lynne Hocking, Chris Holmes, Eleanor Howard, Philip Howard, Joanna M.M. Howson, Debbie Hughes, Sarah Hunt, John D. Isaacs, Mahim Jain, Derek P. Jewell, Toby Johnson, Jennifer D. Jolley, Ian R. Jones, Lisa A. Jones, George Kirov, Cordelia F. Langford, Hana Lango-Allen, G. Mark Lathrop, James Lee, Kate L. Lee, Charlie Lees, Kevin Lewis, Cecilia M. Lindgren, Meeta Maisuria-Armer, Julian Maller, John Mansfield, Jonathan L. Marchini, Paul Martin, Dunecan Co Massey, Wendy L. McArdle, Peter McGuffin, Kirsten E. McLay, Gil McVean, Alex Mentzer, Michael L. Mimmack, Ann E. Morgan, Andrew P. Morris, Craig Mowat, Patricia B. Munroe, Simon Myers, William Newman, Elaine R. Nimmo, Michael C. O'Donovan, Abiodun Onipinla, Nigel R. Ovington, Michael J. Owen, Kimmo Palin, Aarno Palotie, Kirstie Parnell, Richard Pearson, David Pernet, John Rb Perry, Anne Phillips, Vincent Plagnol, Natalie J. Prescott, Inga Prokopenko, Michael A. Quail, Suzanne Rafelt, Nigel W. Rayner, David M. Reid, Anthony Renwick, Susan M. Ring, Neil Robertson, Samuel Robson, Ellie Russell, David St Clair, Jennifer G. Sambrook, Jeremy D. Sanderson, Stephen J. Sawcer, Helen Schuilenburg, Carol E. Scott, Richard Regan Scott, Sheila Seal, Sue Shaw-Hawkins, Beverley M. Shields, Matthew J. Simmonds, Debbie J. Smyth, Elilan Somaskantharajah, Katarina Spanova, Sophia Steer, Jonathan Stephens, Helen E. Stevens, Kathy Stirrups, Millicent A. Stone, David P. Strachan, Zhan Su, Deborah P.M. Symmons, John R. Thompson, Wendy Thomson, Martin D. Tobin, Mary E. Travers, Clare Turnbull, Damjan Vukcevic, Louise V. Wain, Mark Walker, Neil M. Walker, Chris Wallace, Margaret Warren-Perry, Nicholas A. Watkins, John Webster, Michael N. Weedon, Anthony G. Wilson, Matthew Woodburn, B. Paul Wordsworth, Chris Yau, Allan H. Young, Eleftheria Zeggini, Matthew A. Morris J. Brown, Paul R. Burton, Mark J. Caulfield, Alastair Compston, Martin Farrall, Stephen C.L. Gough, Alistair S. Anita Hall, Andrew T. Hattersley, Adrian V.S. Hill, Christopher G. Mathew, Marcus Pembrey, Jack Satsangi, Michael R. Stratton, Jane Worthington, Matthew E. Hurles, Audrey Duncanson, Willem H. Ouwehand, Miles Parkes, Nazneen Rahman, John A. Todd, Nilesh J. Samani, Dominic P. Kwiatkowski, Mark I. McCarthy, Nick Craddock, Panos Deloukas, Peter Donnelly, Jenefer M. Blackwell, Elvira Bramon, Juan P. Casas, Aiden Corvin, Janusz Jankowski, Hugh S. Markus, Colin Na Palmer, Robert Plomin, Anna Rautanen, Richard C. Trembath, Ananth C. Viswanathan, Nicholas W. Wood, Chris C.A. Spencer, Gavin Band, Céline Bellenguez, Colin Freeman, Garrett Hellenthal, Eleni Giannoulatou, Matti Pirinen, Richard Pearson, Amy Strange, Hannah Blackburn, Suzannah J. Bumpstead, Serge Dronov, Matthew Gillman, Alagurevathi Jayakumar, Owen T. McCann, Jennifer Liddle, Simon C. Potter, Radhi Ravindrarajah, Michelle Ricketts, Matthew Waller, Paul Weston, Sara Widaa, and Pamela Whittaker. Comparative performances of machine learning methods for classifying Crohn Disease patients using genome-wide genotyping data. *Sci. Rep.*, 9(1), dec 2019. ISSN 20452322. doi: 10.1038/s41598-019-46649-z.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, volume 9351, pages 234–241. Springer Verlag, 2015. ISBN 9783319245737. doi: 10.1007/978-3-319-24574-4_28.

Thomas Sakoparnig, Chris Field, and Erik van Nimwegen. Whole genome phylogenies reflect long-tailed distributions of recombination rates in many bacterial species. *bioRxiv*, page 601914, 2019. doi: 10.1101/601914. URL https://www.biorxiv.org/content/10.1101/601914v1.

Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.* International Conference on Learning Representations, ICLR, sep 2014. URL http://www.robots.ox.ac.uk/http://arxiv.org/abs/1409.1556.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. In *2nd Int. Conf. Learn. Represent. ICLR 2014 - Work. Track Proc.* International Conference on Learning Representations, ICLR, 2014. URL http://code.google.com/p/cuda-convnet/.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.*, 15:1929–1958, 2014. ISSN 15337928. doi: 10.1214/12-AOS1000.

C. K. Stover, X. Q. Pham, A. L. Erwin, S. D. Mizoguchi, P. Warrener, M. J. Hickey, F. S.L. Brinkman, W. O. Hufnagle, D. J. Kowallk, M. Lagrou, R. L. Garber, L. Goltry, E. Tolentino, S. Westbrock-Wadman, Y. Yuan, L. L. Brody, S. N. Coulter, K. R. Folger, A. Kas, K. Larbig, R. Lim, K. Smith, D. Spencer, G. K.S. Wong, Z. Wu, I. T. Paulsen, J. Relzer, M. H. Saler, R. E.W. Hancock, S. Lory, and M. V. Olson. Complete genome sequence of Pseudomonas aeruginosa PAO1, an opportunistic pathogen. *Nature*, 406(6799):959–964, aug 2000. ISSN 00280836. doi: 10.1038/35023079.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, volume 07-12-June, pages 1–9. IEEE Computer Society, oct 2015. ISBN 9781467369640. doi: 10.1109/CVPR.2015.7298594.

Alex van Belkum, Leah B. Soriaga, Matthew C. LaFave, Srividya Akella, Jean-Baptiste Veyrieras, E. Magda Barbu, Dee Shortridge, Bernadette Blanc, Gregory Hannum, Gilles Zambardi, Kristofer Miller, Mark C. Enright, Nathalie Mugnier, Daniel Brami, Stéphane Schicklin, Martina Felderman, Ariel S. Schwartz, Toby H. Richardson, Todd C. Peterson, Bolyn Hubby, and Kyle C. Cady. Phylogenetic Distribution of CRISPR-Cas Systems in Antibiotic-Resistant Pseudomonas aeruginosa. *MBio*, 6(6), 2015. ISSN 2161-2129. doi: 10.1128/mbio.01796-15.

Bram van Dijk and Paulien Hogeweg. In Silico Gene-Level Evolution Explains Microbial Population Diversity through Differential Gene Mobility. *Genome Biol. Evol.*, 2016. ISSN 17596653. doi: 10.1093/gbe/evv255.

Victoria E. Wagner and Barbara H. Iglewski. P. aeruginosa Biofilms in CF Infection. *Clin. Rev. Allergy Immunol.*, 35(3):124–134, 2008. ISSN 10800549. doi: 10.1007/s12016-008-8079-9.

Jyotsna Talreja Wassan, Haiying Wang, Fiona Browne, and Huiru Zheng. A Comprehensive Study on Predicting Functional Role of Metagenomes Using Machine Learning Methods. *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, 2019. ISSN 15579964. doi: 10.1109/TCBB.2018.2858808.

WHO. Antimicrobial Resistance Fact sheet. *WHO, Antimicrob. Resist.*, 2014.

Zhiqiang Zhang, Yi Zhao, Xiangke Liao, Wenqiang Shi, Kenli Li, Quan Zou, and Shaoliang Peng. Deep learning in omics: a survey and guideline. *Brief. Funct. Genomics*, 18(1):41–57, feb 2019. ISSN 2041-2649. doi: 10.1093/bfgp/ely030. URL https://academic.oup.com/bfg/article/18/1/41/5107348.